



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/735.985	12/15/2003	Man-Pyo Hong	587-34	4178

28249 7590 08/31/2006
DILWORTH & BARRESE, LLP
333 EARLE OVINGTON BLVD.
UNIONDALE, NY 11553

EXAMINER

HOANG, DANIEL L.

ART UNIT PAPER NUMBER

2192

DATE MAILED: 08/31/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/735,985	Applicant(s) HONG ET AL.	
	Examiner Daniel L. Hoang	Art Unit 2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 12/15/03.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1 and 2 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-2 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 15 December 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☒ All b) ☐ Some * c) ☐ None of:
1. ☒ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date <u>12/15/03</u> . | 6) <input type="checkbox"/> Other: _____ |

Detailed Action

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1 and 2 are rejected under 35 U.S.C. 102(e) as being anticipated by Muttik et al. (US Patent No. 6,907,396).

As per claim 1, Muttik teaches:

A method of detecting malicious scripts using code insertion technique, comprising the step of:

checking values related to each sentence belonging to call sequences by using method call sequence detection based on rules including matching rules and relation rules,

wherein the checking step comprises the steps of:

inserting a self-detection routine (malicious behavior detection routine) call sentence before and after a method call sentence of an original script; and

detecting the malicious codes during execution of the script through a self-detection routine inserted into the original script.

*[see column 2, lines 10-27] "A system for emulating computer viruses and/or malicious software that operates by **patching additional program instructions** into an emulator in order to aid in detecting a computer virus and/or malicious software within suspect code. During operation, the system loads a first emulator extension into the emulator. This first **emulator extension includes program instructions** that aid in the process of emulating the suspect code in order to detect a computer virus and/or malicious software. The system also **loads the suspect code into an emulator buffer** within a data space of a computer system. Next, the system performs an emulation using the first emulator extension and the suspect code. This emulation is performed within an insulated environment in the computer system so that the computer system*

Art Unit: 2192

is insulated from malicious actions of the suspect code. During this emulation, the system determines whether the suspect code is likely to exhibit malicious behavior."

[see column 4, lines 42-50] "During the emulation process, emulator extension 204 can read suspect code 108 looking for patterns indicating the suspect code 108 contains a virus or other type of malicious software. Alternatively, emulator extension 204 can set up an environment that is conducive to emulating suspect code 108. For example, emulator extension 204 can configure the system to emulate uncommonly used system calls or opcodes."

[see column 4, lines 53-63] "Emulator extension 204 can be emulated as part of suspect code 108 by patching the emulator extension 204 into suspect code 108, possibly replacing, overlapping or overwriting portions of suspect code 108. In this case, the location where the patching occurs is defined in the database 206. (2) Emulator extension 204 can be executed before the suspect code 108 is executed, which enables emulator extension 204 to set up the environment that emulator extension 204 is responsible for handling."

[see column 5, lines 1-3] "Emulator extension 204 can be emulated after the suspect code 108 is emulated. This allows emulator extension 204 to analyze the results of running the suspect code 108 in order to produce decision 212."

As per claim 2, Muttik teaches:

The method according to claim 1, herein the self-detection routine call sentence is composed of sentences for storing parameters and return values and calling a detection engine, said sentences being inserted before and after the method call sentence when the method call sentence matches with contents described in the matching rule, and wherein the self-detection routine includes a rule-based detection engine for executing for executing the relation rule related to a relevant matching rule when a method corresponding to the matching rule is called and detecting the presence of malicious behavior of the method call sequence, and methods for causing the parameters and return values of the method call sentence satisfying the matching rule to be stored into a buffer usable by the detection engine.

emulator extension includes program instructions [composed of sentences]

emulator extension 204 can read suspect code 108 looking for patterns [matching rules]

emulator extension 204 can set up an environment that is conducive to emulating suspect code [calling detection engine]

Emulator extension 204 can be executed before the suspect code 108 is executed [insert before]

Emulator extension 204 can be emulated after the suspect code 108 is emulated [insert after]

The system also loads the suspect code into an emulator buffer [buffer usable by detection engine]

[see column 4, lines 33-34] "Emulator extension 204 is retrieved from database 206, which contains a plurality of emulator extensions 208, which can be successively loaded into emulator buffer 201 during the emulation process."

[see column 5, lines 37-38] "By using multiple emulator extensions it is possible to deal with conflicting emulator environments. For example, a first emulator extension can configure emulator 110 to detect a virus that is triggered by a system call returning the year 1999, while a

Art Unit: 2192

second emulator extension can configure emulator 110 to detect a virus that is triggered by the same system call returning the year 2000."

Database 206, which examiner is interpreting as a rule-based detection engine, stores a plurality of emulator extensions which depending on the environment, detects the presence of different malicious behavior.

Claim 1 is rejected under 35 U.S.C. 102(b) as being anticipated by Bond et al. (US

Patent No. 6,275,938).

As per claim 1, Bond teaches:

A method of detecting malicious scripts using code insertion technique, comprising the step of:

checking values related to each sentence belonging to call sequences by using method call sequence detection based on rules including matching rules and relation rules,

wherein the checking step comprises the steps of:

inserting a self-detection routine (malicious behavior detection routine) call sentence before and after a method call sentence of an original script; and

detecting the malicious codes during the execution of the script through a self-detection routine inserted into the original script.

[see column 5, lines 24-33] "When an applet such as 362 is to be executed, a host program 36 such as an Internet web browser invokes emulator 39. The emulator employs its own loader module 396 to load the applet code into a predetermined memory area, and to assign another predetermined memory area for its use. These areas are called the "sandbox" for that applet. During execution of the applet, emulator 39 compiles the applet's code in a compiled cache which resides outside the sandbox. During the compilation process, the emulator also inserts the memory sniff code 394 into the cache."

[see column 6, lines 24-27] "Step 421 substitutes the applet's static links with links to thunk modules. That is, emulator 39 finds all calls to APIs 352-354 within the code of applet 362 and changes them to calls to the corresponding thunks 391-393."

[see column 7, lines 1-5] "The thunks for these APIs incorporate the entire code of the corresponding APIs, recompiled so as to execute entirely within the sandbox memory, and capable of allocating memory only within the bounds of the sandbox."

[see column 7, lines 10-14] "Step 422 then compiles the applet's code into object code which can be executed by emulator 39, FIG. 2. Compilation may proceed all at once or by parts as code becomes required; compiled code is placed in a compiled cache 357, FIG. 4, located outside the sandbox."

[see column 7, lines 17-21] "Step 423 inserts check code into the applet's own code to enforce prohibitions against disallowed memory references. This check code, called "sniff code", examines all memory reads and writes made by the applet's code, and allows or disallows them from occurring. "

The following patents and publications are cited to further show the state of the art with respect to detecting malicious code.

US Patent Application No. 20010012214 to Radatti, which is cited to show a method for intercepting, examining and controlling code.

US Patent Application No. 20020056076 to Made, which is cited to show an analytical virtual machine.

US Patent Application No. 20020066024 to Schmall, which is cited to show a method to detect a class of viral code.

US Patent Application No. 20020073323 to Jordan, which is cited to show a method of detecting suspicious privileged access to restricted computer resources.

US Patent Application No. 20020073330 to Chadnani, which is cited to show a method to detect polymorphic script language viruses.

US Patent Application No. 20020083334 to Rogers, which is cited to show detection of viral code using emulation of operating system functions.

US Patent Application No. 20020174349 to Wolff, which is cited to show detecting malicious alteration of stored computer files.

US Patent Application No. 20020178375 to Whittaker, which is cited to show a system for protecting against malicious mobile code.

US Patent Application No. 20050154900 to Muttik, which is cited to show detecting malicious computer program activity using external program calls with dynamic rule sets.

US Patent Application No. 20050204150 to Peikari, which is cited to show an attenuated computer virus vaccine.

US Patent No. 6289455 to Kocher, which is cited to show method for preventing piracy of digital content.

US Patent No. 6775780 to Muttik, which is cited to show a method to detect malicious scripts by analyzing patterns of system calls degenerated during emulation.

Art Unit: 2192

US Patent No. 6785818 to Sobel, which is cited to show thwarting malicious registry mapping modifications.

US Patent No. 7093239 to van der Made, which is cited to show a computer immune system and method for detecting unwanted code in a computer system.

- *. Any response to this Office Action should be **faxed to (571) 273-8300 or mailed to:**

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Hand-delivered responses should be brought to

Customer Service Window
Randolph Building
401 Dulaney Street
Alexandria, VA 22314

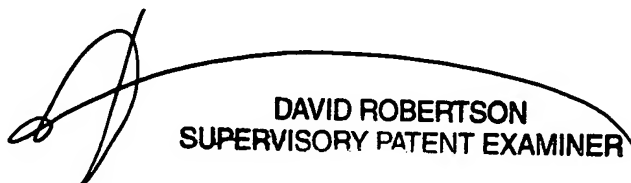
- *. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Daniel L. Hoang whose telephone number is 571-270-1019. The examiner can normally be reached on Monday - Thursday, 8:00 a.m. - 5:00 p.m., EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David Robertson can be reached on 571-272-4186. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



Daniel L. Hoang
8/22/06



DAVID ROBERTSON
SUPERVISORY PATENT EXAMINER